**Lab01: Introduction to algorithm design**

**Capacidades**

* Identificar la importancia de los algoritmos en programación.

**Seguridad**

* Generar un ambiente seguro.
* Evitar el consumo de alimentos.
* Dejar el ambiente ordenado y limpio.

**Preparación**

* El alumno debe revisar previamente el material cargado.

**Recursos**

* Computadora.

**Instrucciones**

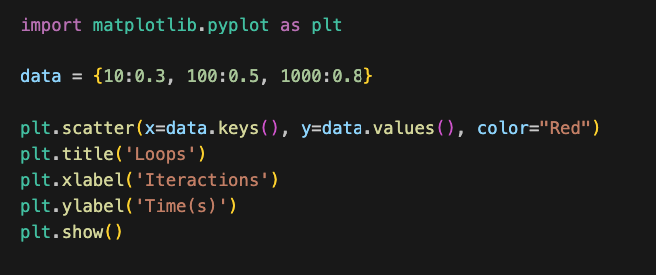
Cada integrante del grupo debe seleccionar un ejercicio diferente y desarrollarlo con la siguiente estructura:

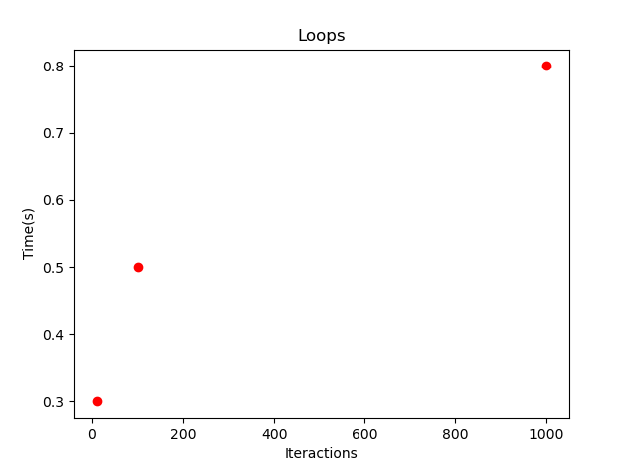
* Nombre del alumno
* Prompt engineering
  + Prompt ingresado
  + Análisis del prompt
  + Ajustes del prompt
  + Comentarios de los compañeros
* Código
  + Código desarrollado
  + Análisis del código
  + Comentarios de los compañeros

Desarrollar todo el código en inglés

**Bibliotecas**

* <https://matplotlib.org/stable/users/installing/index.html>

****

****

**Ejercicios**

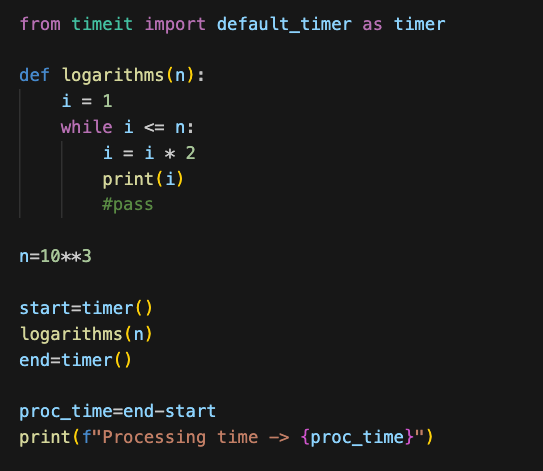
Se debe realizar los ejercicios planteados y un gráfico del tiempo de procesamiento versus los valores de n.

**Cálculo de tiempo de ejecución**

* **Logarithmic complexity - O(log n)**



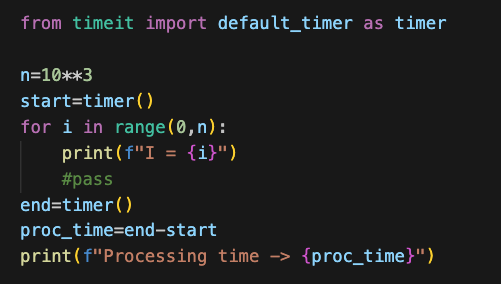
Calcular el tiempo de procesamiento para un condicional con un bucle simple, los valores de n serán : 1,10, 100, 1000, 10000 , 100000, 1000000.



* **Simple Loop - O(n)**



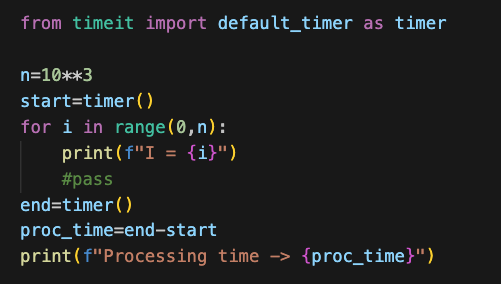
Calcular el tiempo de procesamiento para un bucle simple, los valores de n serán: 10^2, 10^3, 10^4, 10^5 y 10^6.



* **If-then-else statements - O(n)**



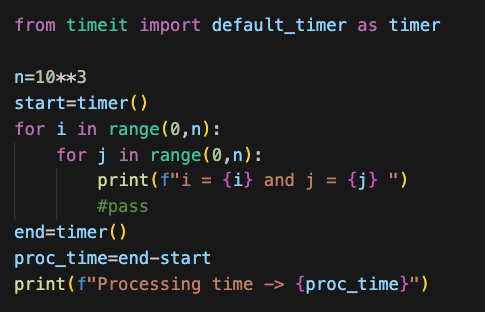
Calcular el tiempo de procesamiento para un condicional con un bucle simple, los valores de n serán: 1, 10, 100, 1000, 10000, 100000.



* **Nested Loops - O(n²)**



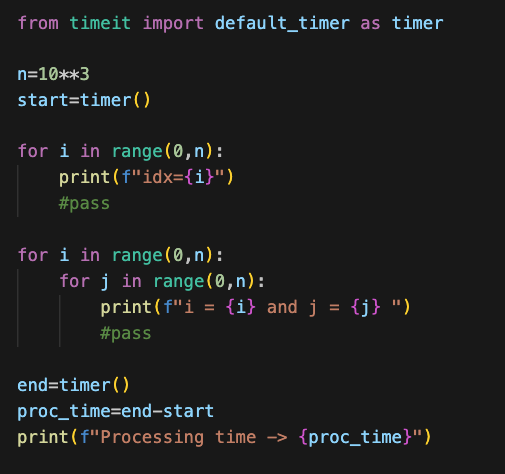
Calcular el tiempo de procesamiento para un bucle anidado de nivel 2, los valores de n serán: 100, 400, 600, 800, 1000, 1100.



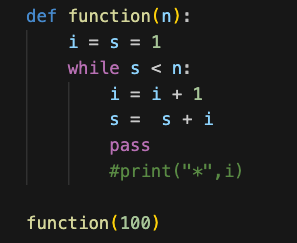
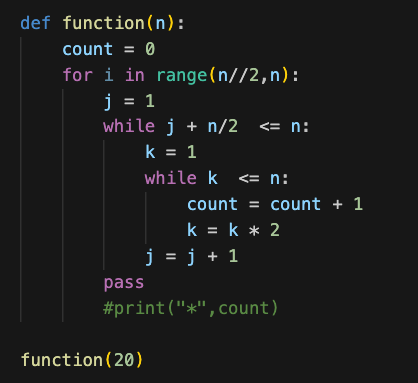
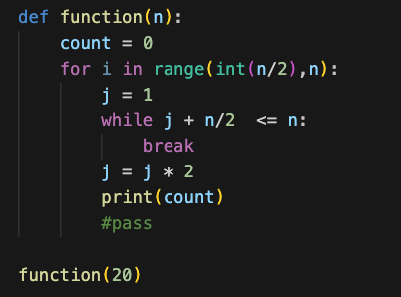
* **Consecutive statements - O(n + n²) = O(n²)**



Calcular el tiempo de procesamiento para un bucle simple unido a un bucle anidado de nivel 2, los valores de n serán: 100, 400, 600, 800, 1000, 1100.



**Identificar algoritmo**

* Algoritmo 1: n = 1, 10, 100, 1000, 10000  
  
* Algoritmo 2: n= 1, 10, 100, 1000   
  
* Algoritmo 3: n = 1,5,10,50,100,500,1000,5000,10000,50000,100000  
  

**Desarrollo**

**Incrementos**

* **Repositorio**
* **Código fuente**
* **Informe**

**Conclusiones**